REMARKS

Reconsideration of the above-identified application in view of the amendments above and the remarks following is respectfully requested.

Claims 18-37 are in this case.

Claims 20 and 26 were rejected under 35 U.S.C §112, first paragraph.

Claims 20,22, 23-28 were rejected under 35 U.S.C §112, second paragraph.

Claims 18-37 were rejected under 35 U.S.C §102(e) as being anticipated by Candelore et al. US patent 6,061,449 (hereinafter Candelore)

While continuing to traverse the Examiner's rejections, and without in any way prejudicing the patentability of the rejected claims, the Applicant has, in order to expedite the prosecution, chosen to amend the claims thereby rendering moot Examiner's rejections.

Rejections under 35 U.S.C § 112, first paragraph

Claims 20 was rejected under 35 U.S.C §112, first paragraph as failing to comply with the written description requirement. Claim 20 includes the element *wherein said executing includes self-decrypting said modified executable program file and secured digital content accessible by said executable program file.* According to Examiner, the present specification does not provide support for "self-decryption" of a file and of digital content. Applicant respectfully traverses Examiner's rejection. Although the term "self-decryption" does not have literal support, support for "self-decryption" is implicit and inherent throughout the specification. The following exemplary paragraph of the present application and the drawing of Figure 2 implicitly describe "self decryption" (and "self-encryption") both of the modified executable program file and digital content accessible to the modified executable program file.

> [0065] FIG. 2 illustrates a preferable structure of a mine 250, according to one embodiment of the invention. Steps that are shown in these figures in dotted lines are optional. Referring to FIG. 2, in step 200 the operation of the mine begins. In the optional step 201, the mine detects the signature key and decrypts the following steps of the mine if they are encrypted. In step 202, the mine uses one or more keys from other mines to decrypt successive steps of the mine. In the optional step 203, the mine decrypts, following steps of the mine using an authentication key or a content key (when a content is to be protected). In step 204, the mine uses one or more of the above keys to decrypt encrypted fragment 205 of the protected software. This fragment 205 is

> actually the software fragment that the author chooses to protect. It is not possible to run the protected software fragment without executing the mine, i.e., passing the entire authentication and decryption procedures of the mine. After the decrypted software fragment by the mine 250, in step 206, the mine re-encrypts the software fragment 205, and the mine itself, by the same key/s as were used for its decryption. Then, the procedure continues in step 207, until reaching another mine.

While continuing to traverse Examiner's rejection of Claim 20, Applicant has chosen to amend claim 20 herein based on paragraph [0065] cited above and thereby more explicitly claim the notion of "self-decryption", rendering moot Examiner's rejection.

Claim 26 was rejected because of the phrase "signature key authenticates content" is not clear. Claim 26 is amended herein removing the objectionable phrase, rendering moot Examiners rejection.

Rejections under 35 U.S.C §112, second paragraph

Examiner objected to references to "executing" in claim 18, wherein two instances of the term "executing" are found. Claim 18 has been amended so that the second instance of "executing" is amended to "running" thereby removing any possible ambiguity. Claim 37 is similarly amended herein.

Claim 22 is amended according to Examiners suggestion to "said executable program file".

Rejections under 35 U.S.C §102

The References and Differences of the Present Invention Thereover:

Prior to discussing the claims, Applicant will first discuss the references of the prior art of record and the novelty of the present invention and its unobviousness over the references.

By way of introduction, Applicant respectfully affirms that there are fundamental differences between Candelore and the present invention. The disclosure according to Candelore is based on dedicated hardware ("secure circuit", column 1 lines 32-47), Figure 1 Candelore). Thus, protecting software according to Candelore

et al. requires software users to have a computer system with a specific secure processor.

The processor secure circuit of Candelore performs a well known technique known as "cipher block chaining". In the context of the present invention, Candelore is virtually identical to Lie et al., "Architectural Support for Copy and Tamper Resistant Software", (hereinafter Lie) cited by the Examiner in the previous office action as both references disclose "cipher block chaining" using a dedicated processor. Applicant repectfully submits, as in the previous office action that Candelore is not any more relevant to the patentability of the present invention than is Lie. Many of the arguments of the previous office action response are reiterated herein.

In cipher-block chaining, the data is encrypted block by block. The blocks are necessarily of equal size, for instance of 8 bytes (see Candelore column 6 line 50). Each block of plaintext is XORed with the previous ciphertext block before being encrypted. This way, each ciphertext block is dependent on all plaintext blocks up to that point. To make each message unique, an initialization vector typically dependent on the processor is used in the first block.

Applicant has very carefully reviewed Candelore, and has not found any reference or suggestion to steps of the present invention: *designating a critical locations within the executable program file*; and *arming the executable program file, by including software procedures or mines at the location*s. Applicant has especially reviewed column 28 lines 27-38 (as pointed out in present Office Action page 5 line 20) Candelore is simply pointing out in (28:27-38) that while using a cipher block chaining technique, it is efficient to divide the data into blocks so that instructions are not broken up between adjacent blocks. For instance, if an instruction for a particular "secure" processor has instructions of 4 bytes, dividing the data to be encrypted into blocks of 8 bytes allows two instructions to fit perfectly into each block. Hence, the decoding of each block is completed without waiting for the decryption of the next block.

Applicant respectfully submits that neither Candelore nor Lie nor any other reference in the field of secure processors and cipher block chaining teaches, suggests or implies *designating a critical locations within the executable program file*; and *arming the executable program file, by including software procedures or mines at the location*s. In cipher block chaining, each data file of for instance 80,000 bytes is

divided automatically into 10,000 blocks, each block of 8 bytes. Hence, in cipher block chaining there is no step analogous to *designation of critical locations* within the data. In cipher block chaining, there is no designation of critical locations in the program file and/or digital content to be encrypted. In fact, disclosures in the field of "cipher block chaining" teach away from the present invention. In cipher block chaining, the division into blocks is performed blindly and uniformly without considering the criticality of the locations within the data to be encrypted.

Regarding the limitation " *arming the executable program file, ... by including ... software procedures at said locations wherein each said software procedure performs at least one linked portion of the securing* ", Applicant has carefully reviewed Candelore, especially column 25, lines 20-28 and 50-58 and the associated Figure 2. Applicant respectfully traverses Examiner's finding that Candelore discloses or even suggests *arming.. by including software procedures* [mines] at the critical locations. Candelore is describing features well known in "cipher block encryption" wherein the encryption of each block of data is dependent on the block being encrypted as well as the previous blocks. There is no mention or suggestion in Candelore of *arming , i.e.* adding *software procedures* or mines, interspersed into the data being encrypted. Furthermore, the notion of *arming* would be considered inoperable or at least impractical in the context of cipher block encryption by a person having ordinary skill in the art.

Similarly, Applicant has carefully reviewed all prior art of record and found none which are relevant at all to the present invention.

Independent Claims 18 and 37

Independent claims 18 and 37 include novel physical features or steps. Specifically, "*designating a plurality of critical locations within the executable program file*; *arming the executable program file..... by including a plurality of software procedures at said critical locations..., ...running said software procedures solely upon reaching said respective locations,* are novel features not previously disclosed or even suggested in the prior art. Applicant respectfully traverses Examiners finding that Candelore, Lie or any other reference in the field of secure processors and/or cipher block encryption anticipate the present invention. In line with this are the following rulings:

> A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987)
>
> To serve as an anticipation when the reference is silent about the asserted inherent characteristic, such gap in the reference may be filled with recourse to extrinsic evidence. Such evidence must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill." *Continental Can Co. USA v. Monsanto Co.*, 948 F.2d 1264, 1268, 20 USPQ2d 1746, 1749 (Fed. Cir. 1991)
>
> Extrinsic evidence may be used to explain but not expand the meaning of terms and phrases used in the reference relied upon as anticipatory of the claimed subject matter. *In re Baxter Travenol Labs.*, 952 F.2d 388, 21 USPQ2d 1281 (Fed. Cir. 1991)

Novel physical features of Independent claims 18 and 37

Produce New and Unexpected Results

The term "mines" or *software procedures* of the present invention, (as defined on page 8 of the present application) perform a security function such as checking the authenticity and/or validity of data in use. Proper operation/use of the present invention depends on one or more keys: validation key , authentication key and/or signature key which enable additional functionality (*e.g.* authentication and validation of the data). The additional functionality of a mine differentiates the present invention from Candelore, Lie or any other references based on cipher block chaining which does not include any additional functionality and is designed only to prevent relocation of code segments by an adversary.

Regarding the limitation "*executing*", the guarding module of the prior art, in contrast runs constantly in the background while a "mine" or "software procedure"of the present invention does not run in the background and is activated when the mine is reached during execution of the protected software or content.

Applicant wishes to point out "new and unexpected results" of using distinct steps d*esignating* and a*rming*. *Designating* is used just to mark the software parts to be protected and generate a flagged software file. The flagged software doesn't include any additional functionality comparing to the original un-flagged software. At a later stage, during the *arming* step, a machine code containing security functionality is

inserted into the marked locations at the flagged software . The use of two distinct steps *designating and arming* is essential for commercial utilization, because the provider of the copy protection (*e.g.* a CD replication facility) doesn't wish to grant to the software vendor, author or copyright owner fully functional tools that can be used for an unlimited number of copies. Instead, the tools delivered to the software vendor do not add any functionality, but just mark the locations in the software to be protected. The copy protection provider can then perform the arming process on the number of copies requested by the software vendor (*e.g.* replicating 1000 CDs containing the armed software) and charge the software vendor for the copy protection service according to the number of copies on which the arming process was performed. This process is illustrated in figures 1d and 1e.

The present invention is a method for authenticating and protecting digital data from illegal copy and use, which is independent of the processor or any other hardware. Thus, protected software according to the present invention can be executed on any computer hardware. Moreover, software vendors that wish to protect their software, according to cipher block encryption (Candelore, Lie), must encrypt each copy of the software/content for the specific processor of each user. All the recipients of the software must be known (or at least their processors) and each copy of the software must be delivered to a specific recipient. A software copy delivered to a specific user will immediately become useless when the user replaces/upgrades his processor or replaces his entire computer system. The rapid enhancements nowadays in computer systems and processors especially, cause software consumers to upgrade their computer systems very frequently, rendering impractical the protection of software using cipher block encryption methods such as Candelore, or Lie. Furthermore, many software/content consumers own more than one computer system. Software protected according to Candelore (or Lie) can be executed/decrypted only by a specific processor, rendering the disclosure Candelore impractical for software vendors. Some software is designed to work on multiple computers (clusters of computers). For example, to increase performance by balancing the load of processing tasks on several computers. Sometimes each computer may have a dedicated task such as sound processing or video processing, and in other cases, the same code is required to run on several different computers. Since the disclosure of Candelore (or Lie) requires that the software code will be encoded specifically for each processor this task becomes too complex and impractical.

According to the present invention, software vendors that wish to protect their software according to the present invention, are not limited in their distribution methods. For example, unlimited number of copies of the software can be produced and distributed via retail stores regardless of the computer in use by the end user. The receiver of the protected software or content may execute the software or present the content on any computer system as long as the software/content is stored on the original medium.

Dependent Claims

Although Applicant submits that independent claims 18 and 37 include novel steps and are not obvious, thereby rendering all dependent claims therefrom also patentable, Applicant wishes to briefly point out at least some of the patentable aspects of dependent claims in their own right.

Applicant respectfully traverses Examiner's rejection of claim 26. In the present invention, signature keys and content keys are stored together with the protected data on the same medium. Therefore the owner of the specific original medium can transport the medium to any other computer system and use the protected data based on his sole decision without requiring permission such as a certificate from "a trusted certificate authority" *e.g.* VeriSign. The present invention can also be utilized using a user authentication key for even removing the dependency of the specific medium. These properties make the present invention much more flexible and versatile than cipher block chaining and other prior art which do not store signature and content keys on the same medium.

Applicant respectively traverses Examiner's rejection of claim 23. In Candelore, Lie and other references using "secure processors", encrypted software is run on a specific processor by decrypting the software with the processor's private key. According to the present invention, an authentication key allows operation of the protected software. Furthermore, different software vendors, can allow a specific user to run their software at the same time on a single processor using a different authentication key accessible via the Internet. This cannot be achieved using cipher block encryption and a dedicated processor.

Commercial Success as Secondary Consideration showing
New and Unexpected Results

Applicant wishes to point out that the present invention, for the reasons stated previously, has been successfully utilized on millions of copies of commercial software and other digital content published by major publishers around the world as well as government and military organizations. The present invention is utilized successfully in protecting all kinds of digital information carried on various media types such as CD-ROM, recordable discs and the Internet. The present invention is commercially available worldwide via HexaLock Ltd (http://www.hexalock.com).

Applicant respectfully requests Examiner to reconsider declaration under 37 CFR 1.132 filed December 14, 2005 in light of the remarks presented herein.

In view of the above amendments and remarks it is respectfully submitted that independent claims 18 and 37 and claims dependent therefrom are in condition for allowance. Prompt notice of allowance is respectfully and earnestly solicited.

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: Aug 10, 2006